Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	12875	707/100, or 707/7 or 707/200 or 707/101 or 707/100 or 707/3	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:34
L2	50	1 and (sorted near2 tree)	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:34
L3	19	2 and ((empty near2 node) or NULL)	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:34
L4	0	2 and ((empty near2 node) or NULL) and (redistribute same nodes)	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:43
L5	11	1 and (redistribute same nodes)	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:35
L6	1	1 and (redistribute same nodes) and (empty near2 (space or node))	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:35
L7	1	1 and (redistribute same nodes) and (empty near2 (spaces or nodes))	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:38
L8	1	1 and (redistribute same nodes) and (empty near2 (spaces or nodes)) and index	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:38
L9	1	1 and (redistribute same nodes) and (empty near2 (spaces or nodes)) and index and pointer	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:39

L10	1	1 and (redistribute same nodes) and (empty near2 (spaces or nodes)) and index and pointer and (empty)	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:39
L11	13837	1 or 707/201	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:40
L12	0	11 and (redistribute same (empty near2 (nodes or spaces)))	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:41
L13	0	(redistribute same (empty near2 (nodes or spaces)))	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:40
L14	0	1 and (redistribute same (empty near2 (nodes or spaces)))	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:41
L15	74	1 and (empty near2 (nodes or spaces))	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:41
L16	47	15 and tree	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:41
L17	28	15 and tree and (data near2 structure)	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:41
L18	5	15 and tree and (data near2 structure) and (sorted with tree)	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:41
L19	5	15 and tree and (data near2 structure) and (sorted with tree) and (pointer or link)	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:41

L20	0	15 and tree and (data near2 structure) and (sorted with tree) and (pointer or link) and redistribute	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:42
L21	4	15 and tree and (data near2 structure) and (sorted with tree) and (pointer or link) and insert	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:42
L22	2	15 and tree and (data near2 structure) and (sorted with tree) and (pointer or link) and insert and delete	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:43
L23	22	((empty near2 node) or NULL) and (redistribute same nodes)	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:43
L24	15	((empty near2 node) or NULL) and (redistribute same nodes) and insertion	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:44
L25	1	((empty near2 nodes)) and (redistribute same nodes) and insertion	USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/31 09:44

Web Images Groups News Froogle Local[New!] **more »**

redistribute a tree containing empty nodes Search Advanced Search Preferences

Web Results **1** - **10** of about **101,000** for **redistribute a tree containing empty nodes**. (**0.35** seconds)

B-**tree** algorithms
A B-**tree** consists of "**node**" records **containing** the keys, and pointers that ...
If a **node** underflows, we may be able to "**redistribute**" keys by borrowing some ...
www.semaphorecorp.com/btp/algo.html - 11k - Oct 29, 2005 - Cached - Similar pages

search.cpan.org: Pod::**Tree** - Create a static syntax **tree** for a POD
Creates a new Pod::**Tree** object. The syntax **tree** is initially **empty**. ... Returns the root **node** of the syntax **tree**. See Pod::**Tree**::**Node** for a description of ...
search.cpan.org/~swmcd/Pod-Tree-1.11/Tree.pm - 14k - Cached - Similar pages

Copyright 1999 by Steven McDougall. This module is free # software ...
The syntax **tree** is initially **empty**. ... See L<Pod::**Tree**::**Node**> for a description of the syntax **tree**. =item I<**$tree**>->C<dump> Pretty prints the syntax **tree**. ...
search.cpan.org/src/SWMCD/Pod-Tree-1.00/Tree.pm - 9k - Cached - Similar pages
[More results from search.cpan.org]

[RTF] CX214 Balanced Trees
File Format: Rich Text Format - View as HTML
(a) If a sibling has 2 items, **redistribute** values. **Node** n adopts a child from
... Deletion: **nodes** merge when **empty**. Red-Black Trees. A red-black **tree** ...
community.middlebury.edu/~briggs/ Courses/CX214-S02/CX214-balanced-**trees**.rtf - Similar pages

[PDF] 7.3 SELF-ADJUSTING BINARY SEARCH TREES
File Format: PDF/Adobe Acrobat - View as HTML
then K is in **node** P, and otherwise P has an **empty** child where the search for K
... We regard each **node** of the **tree** as a bank account **containing** a certain ...
www.ida.liu.se/~TDDB56/SplaytreeChapter/Chapter_7_3.pdf - Similar pages

Pod::**Tree**
The syntax **tree** is initially **empty**. $ok = **$tree**-> load_file ($file, %options)
... See Pod::**Tree**::**Node** for a description of the syntax **tree**. ...
world.std.com/~swmcd/steven/ perl/lib/Pod/Tree/Pod/Tree.html - 10k - Cached - Similar pages

[PDF] Randomized Binary Search Trees
File Format: PDF/Adobe Acrobat - View as HTML
The **empty**. **tree** or external **node** is denoted by . Besides the definition of random
... Hence, the **tree containing** the keys smaller than 3 in the original ...
www.lsi.upc.es/~conrado/research/papers/jacm-mr98.pdf - Similar pages

[PS] 6.897: Advanced Data Structures Spring 2003 Lecture 14 – Monday ...
File Format: Adobe PostScript - View as Text
Conceptually, each **node** of the **tree** represents an interval which is the ...
2.6 Density The density of a **node** represents how full or how **empty** the interval ...
theory.csail.mit.edu/classes/ 6.897/spring03/scribe_notes/L14/lecture14.ps - Similar pages

[PS] An Evaluation of Software Cacheing in Astrophysical n-body ...
File Format: Adobe PostScript - View as Text
In this case the internal **nodes** of the **tree** represent a volume of space with an
... If the "bottom list" is **empty**, the only **node** at the bottom level of the ...
www.doc.ic.ac.uk/~ajf/Research/Papers/nbody/PCW.ps - Similar pages

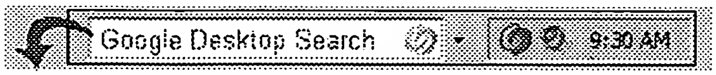
DTView Documentation
Alternatively, the name of file **containing** the decision or regression **tree** to

display ... For all other inner **nodes** and for all leaves the upper label field ...
fuzzy.cs.uni-magdeburg.de/ ~borgelt/doc/dtview/dtview.html - 25k - Cached - Similar pages

Google Groups results for **redistribute a tree containing empty nodes**

AVL **tree** library (part 2 of 2) - alt.sources - Mar 28, 1991
v27i033: AVL **Tree** subroutines (replaces v11i020 from ... - comp.sources.unix - Sep 06, 1993

Gooooooooogle ▶
Result Page: 1 2 3 4 5 6 7 8 9 10 **Next**



redistribute a tree containing empty | Search

Search within results | Language Tools | Search Tips | Dissatisfied? Help us improve

Google Home - Advertising Programs - Business Solutions - About Google

Set	Items	Description
S1	577196	TREE OR TREES OR BTREE OR DIRECTORY OR DIRECTORIES OR TRIE OR TRIES
S2	1678973	NODE? OR BRANCH? OR LEAF? OR JUNCTION? OR JUNCTURE? OR INT- ERSECT?
S3	16824	S2(2N)(ADJOIN? OR NEXT OR PRIOR OR PREVIOUS OR FOLLOWING OR PARALLEL OR CONTIGUOUS? OR CONNECTING? OR PARENT()CHILD? OR - ORDINATE(N)SUBORDINAT? OR LINKED OR SEQUENTIAL?)
S4	136093	EMPTY? OR UNUSED? OR UNFILLED OR "NOT"(N)(FULL OR USED OR - USE OR FILLED)
S5	9722474	REARRANG? OR REORDER? OR RESORT? OR REDISTRIBUT? OR DISTRI- BUT? OR INSERT OR INSERTING OR INSERTS OR ORDER? OR ARRANG? OR SWAP? OR REVERS?
S6	2808663	DATASTRUCTUR? OR DATA()(ELEMENTS OR OBJECT OR OBJECTS OR S- TRUCTUR? OR ITEMS) OR STACK? OR ARRAY? OR TREE OR BTREE OR MA- TRIX?
S7	1855762	END OR ENDPOINT OR ENDS OR TERNIMAL OR TERMINUS OR LEAF? OR LEAVES OR TAIL OR TAILS
S8	6	S3 AND S4 AND S5 AND S6
S9	141	S3 AND S5 AND S6 AND S7
S10	62	S2 AND S4 AND S5 AND S6 AND S7
S11	104	S9 AND S1
S12	17	S11 AND (PAIR OR TWO OR SECOND OR 2ND OR ANOTHER OR OTHER - OR TWIN OR DUO OR DUAL OR BOTH)(2N)S2
S13	85	S8 OR S10 OR S12
S14	62	RD (unique items)
S15	50	S14 NOT PY>2001

File 8:Ei Compendex(R) 1970-2005/Aug W3
(c) 2005 Elsevier Eng. Info. Inc.
File 35:Dissertation Abs Online 1861-2005/Aug
(c) 2005 ProQuest Info&Learning
File 56:Computer and Information Systems Abstracts 1966-2005/Aug
(c) 2005 CSA.
File 57:Electronics & Communications Abstracts 1966-2005/Aug
(c) 2005 CSA.
File 65:Inside Conferences 1993-2005/Aug W4
(c) 2005 BLDSC all rts. reserv.
File 2:INSPEC 1969-2005/Aug W3
(c) 2005 Institution of Electrical Engineers
File 94:JICST-EPlus 1985-2005/Jul W1
(c)2005 Japan Science and Tech Corp(JST)
File 111:TGG Natl.Newspaper Index(SM) 1979-2005/Aug 31
(c) 2005 The Gale Group
File 6:NTIS 1964-2005/Aug W3
(c) 2005 NTIS, Intl Cpyrght All Rights Res
File 144:Pascal 1973-2005/Aug W3
(c) 2005 INIST/CNRS
File 434:SciSearch(R) Cited Ref Sci 1974-1989/Dec
(c) 1998 Inst for Sci Info
File 34:SciSearch(R) Cited Ref Sci 1990-2005/Aug W4
(c) 2005 Inst for Sci Info
File 62:SPIN(R) 1975-2005/Jun W4
(c) 2005 American Institute of Physics
File 99:Wilson Appl. Sci & Tech Abs 1983-2005/Jul
(c) 2005 The HW Wilson Co.
File 95:TEME-Technology & Management 1989-2005/Jul W4
(c) 2005 FIZ TECHNIK

15/5/4 (Item 4 from file: 8)

04623672 E.I. No: EIP97023519619
Title: On-line reorganization of sparsely-populated B plus -trees**
Author: Zou, Chendong; Salzberg, Betty
Corporate Source: Northeastern Univ, Boston, MA, USA
Conference Title: Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data
Conference Location: Montreal, Can Conference Date: 19960604-19960606
Sponsor: ACM SIGMOD
E.I. Conference No.: 45963
Source: SIGMOD Record (ACM Special Interest Group on Management of Data) v 25 n 2 June 1996.. p 115-124
Publication Year: 1996
CODEN: SRECD8
Language: English
Document Type: CA; (Conference Article) Treatment: G; (General Review)
Journal Announcement: 9704W1
Abstract: In this paper, we present an efficient method to do on-line reorganization of sparsely-populated B** plus -trees. It reorganizes the **leaves** first, compacting in short operations groups of **leaves** with the same parent. After compacting, optionally, the new **leaves** may **swap** locations or be moved into **empty** pages so that they are in key **order** on the disk. After the **leaves** are reorganized, the method shrinks the **tree** by making a copy of the upper part of the **tree** while leaving the **leaves** in place. A new concurrency method is introduced so that only a minimum number of pages are locked during reorganization. During **leaf** reorganization, Forward Recovery is used to save all work already done while maintaining consistency after system crashes. A heuristic algorithm is developed to reduce the number of **swaps** needed during **leaf** reorganization, so that better concurrency and easier recovery can be achieved. A detailed description of switching from the old B** plus - **tree** to the new B** plus - **tree** is described for the first time. (Author abstract) 16 Refs.
Descriptors: *Database systems; Online systems; Concurrency control; Heuristic methods; Algorithms; Trees (mathematics)
Identifiers: **Leaf** reorganization; Forward recovery
Classification Codes:
723.3 (Database Systems); 722.4 (Digital Computers & Systems); 921.4 (Combinatorial Mathematics, Includes Graph Theory, Set Theory)
723 (Computer Software); 722 (Computer Hardware); 921 (Applied Mathematics)
72 (COMPUTERS & DATA PROCESSING); 92 (ENGINEERING MATHEMATICS)

15/5/6 (Item 6 from file: 8)
DIALOG(R)File 8:Ei Compendex(R)

01388695 E.I. Monthly No: EI8309072597 E.I. Yearly No: EI83022822
Title: STORAGE UTILIZATION IN B*-TREES WITH A GENERALIZED OVERFLOW TECHNIQUE.
Author: Kuespert, Klaus
Corporate Source: Univ of Kaiserslautern, Dep of Computer Science, Kaiserlautern, West Ger
Source: Acta Informatica v 19 n 1 Apr 1983 p 35-55
Publication Year: 1983
CODEN: AINFA2 ISSN: 0001-5903
Language: ENGLISH
Journal Announcement: 8309
Abstract: Storage utilization in random B*-trees (trees, where all data stored in the **leaf nodes**) is analyzed. Extending a proposal of R. Bayer and E. McCreight, in case of insertion into a full **node** , up to m MINUS 1 adjacent **nodes** are scanned for **empty** space. If this search is successful, entries are shifted on **leaf** level to gain free space for the new one; otherwise, the entries of the m **nodes** scanned are **distributed** as uniformly as possible over m PLUS 1 **nodes** . Using iterative models it is shown that for large trees of high **order** storage utilization converges to m X (TIMES) ln(m PLUS 1)/m). 16 refs.
Descriptors: *DATA PROCESSING--* **Data Structures**
Classification Codes:
723 (Computer Software)
72 (COMPUTERS & DATA PROCESSING)

15/5/8 (Item 2 from file: 35)
DIALOG(R)File 35:Dissertation Abs Online

01501328 ORDER NO: AAD96-27541
PERFORMANCE STUDY OF CONCURRENT SEARCH TREES AND HASH ALGORITHMS ON MULTIPROCESSOR SYSTEMS
Author: DEMUYNCK, MARIE-ANNE
Degree: PH.D.
Year: 1996
Corporate Source/Institution: UNIVERSITY OF NORTH TEXAS (0158)
Source: VOLUME 57/04-B OF DISSERTATION ABSTRACTS INTERNATIONAL. PAGE 2663. 260 PAGES
Descriptors: COMPUTER SCIENCE
Descriptor Codes: 0984

This study examines the performance of concurrent algorithms for B-trees and linear hashing. B-trees are widely used as an access method for large, single key, database files, stored in lexicographic **order** on secondary storage devices. Linear hashing is a fast and reliable hash algorithm, suitable for accessing records stored unordered in buckets.

This dissertation presents performance results on implementations of concurrent Bi$\sp{link}$- **tree** and linear hashing algorithms, using lock-based, partitioned and **distributed** methods on the Sequent Symmetry shared memory multiprocessor system and on a network of **distributed** processors created with PVM (Parallel Virtual Machine) software. Initial experiments, which started with **empty data structures** , show good results for the partitioned implementations and lock-based linear hashing, but poor ones for lock-based B$\sp{link}$-trees. A subsequent test, which started with loaded **data structures** , shows similar results, but with much improved performances for locked B$\sp{link}$-trees. The data also highlighted the high cost of split operations, which reached up to 70% of the total **insert** time.

To improve the performance of the B- **tree data structure** in a parallel computing environment, we have developed the B$\sp{mad}$- **tree** , a B$\sp{link}$- **tree** variant. It allows insertion without **node** splits, with multiple access in its **leaf nodes** , and dilation in both the index and the **leaf nodes** . Concurrent search, **insert** and restructuring algorithms for partitioned, locked and **distributed** models are given. Two locked approaches are used; both minimize the necessary number of locks. Only part of an insertion **node** is locked during **insert** , and simultaneous insertions by multiple processors in the same **node** are allowed. A restructuring algorithm runs periodically in the background and requires only waits. At most one such wait is encountered by any search or update operation.

The B$\sp{mad}$- **tree** implementations showed very good results for locked and partitioned algorithms. Especially the locked algorithms exceeded expectations. The **distributed** results were disappointing. High communication costs prevented a good performance. Experimental data were used to project performance beyond the current test systems.

This research also prompted some further investigations, such as analyzing the high cost of process creation and the development of a load balancing method.

15/5/12 (Item 6 from file: 35)
DIALOG(R)File 35:Dissertation Abs Online

01138998 ORDER NO: AAD91-02527
THE PERFORMANCE OF CONCURRENT DATA STRUCTURE **ALGORITHMS (B-TREES)**
Author: JOHNSON, THEODORE J.
Degree: PH.D.
Year: 1990
Corporate Source/Institution: NEW YORK UNIVERSITY (0146)
Adviser: DENNIS E. SHASHA
Source: VOLUME 51/08-B OF DISSERTATION ABSTRACTS INTERNATIONAL. PAGE 3937. 250 PAGES
Descriptors: COMPUTER SCIENCE
Descriptor Codes: 0984

This thesis develops a validated of concurrent **data structure** algorithm performance, concentrating on concurrent B-trees. The thesis first develops two analytical tools, which are explained in the next two paragraphs, for the analysis.

Yao showed that the space utilization of a B- **tree** built from random **inserts** is 69%. Assuming that **nodes** merge only when **empty** , we show that the utilization is 39% when the number of **insert** and delete operations is the same. However, if there are just 5% more **inserts** than deletes, then the utilization is at least 62%. In addition to the utilization, we calculate the probabilities of splitting and merging, important parameters for calculating concurrent B- **tree** algorithm performance. We compare merge-at- **empty** B-trees with merge-at-half B-trees. We conclude that merge-at- **empty** B-trees have a slightly lower space utilization but a much lower restructuring rate, making merge-at- **empty** B-trees preferable for concurrent B- **tree** algorithms.

We analyze queues that service readers and writers. Readers are served concurrently and writers are served serially. Customers receive service in FCFS **order** . We show that the additional time that a writer must wait for preceding readers increases logarithmically with the proportion of readers to writers. From this, we can calculate the expected wait in the queue and the capacity of the queue.

We use the analytical tools to model a representative set of concurrent B- **tree** algorithms. The algorithms use a variety of locking, restructuring and path recovery techniques. The analyses, which are validated by simulations, show that the Link-style algorithms are by far the best. The analyses are extended to account for buffering and database recovery. We show that holding non- **leaf** locks until commit time is bad for performance.

The analysis of concurrent B- **tree** algorithms can be extended to cover other concurrent **data structure** algorithms. We describe a template for analyzing concurrent **tree data structures** , and apply the template to some concurrent extensible hashing schemes.

15/5/15 (Item 2 from file: 2)
DIALOG(R)File 2:INSPEC

7012624 INSPEC Abstract Number: C2001-09-6160B-019
Title: Constraint satisfaction for reconciling heterogeneous tree **databases**
Author(s): Kitakami, H.; Nishimoto, M.
Author Affiliation: Hiroshima City Univ., Japan
Conference Title: Database and expert systems applications. 11th International Conference, DEXA 2000. Proceedings (Lecture Notes in Computer Science Vol.1873) p.624-33
Editor(s): Ibrahim, M.; Kung, J.; Revell, N.
Publisher: Springer-Verlag, Berlin, Germany
Publication Date: 2000 Country of Publication: Germany xix+1003 pp.
ISBN: 3 540 67978 2 Material Identity Number: XX-2001-01532
Conference Title: Database and Expert Systems Applications. 11th International Conference, DEXA 2000. Proceedings
Conference Date: 4-8 Sept. 2000 Conference Location: London, UK
Language: English Document Type: Conference Paper (PA)
Treatment: Practical (P)
Abstract: In **order** to simplify the reconciliation of two heterogeneous **tree** databases, we must minimize the number of crossovers in a directed graph constructed using two subtrees selected from the databases. The paper proposes a method for minimizing the number of crossovers in the directed graph. To find the directed graph with the minimum number of crossovers, the method maintains zero-crossovers in each **ordered** subtree. The resulting directed graph is defined as a semi-optimal solution satisfying the zero-crossover constraint for edges **connecting two leaf** sequences. It is computed by changing the **order** of non- **leaf** nodes in each hierarchical level of the **ordered tree** and **swapping leaf** nodes in each of the **two leaf** layers. To maintain the zero-crossover constraint for each **ordered tree** in the **matrix** transformation, the method also finds the **two leaf** clusters that contain half of the **leaf** nodes and **swaps** the **leaf** clusters. (10 Refs)
Subfile: C
Descriptors: constraint theory; directed graphs; **distributed** databases; optimisation; **tree data structures**
Identifiers: constraint satisfaction; heterogeneous **tree** database reconciliation; crossovers; directed graph; minimum crossovers; semi-optimal solution; zero-crossover constraint; **leaf** sequences; non- **leaf** nodes; hierarchical level; **leaf** layers; **matrix** transformation; **leaf** clusters; **leaf** nodes
Class Codes: C6160B (Distributed databases); C4250 (Database theory); C6120 (File organisation); C1160 (Combinatorial mathematics); C1180 (Optimisation techniques)

15/5/17 (Item 4 from file: 2)
DIALOG(R)File 2:INSPEC

5324366 INSPEC Abstract Number: C9609-6120-002
Title: On-line reorganization of sparsely-populated B/sup +/-trees
Author(s): Chendong Zou; Salzberg, B.
Author Affiliation: Coll. of Comput. Sci., Northeastern Univ., Boston, MA, USA
Journal: SIGMOD Record Conference Title: SIGMOD Rec. (USA) vol.25, no.2 p.115-24
Publisher: ACM,
Publication Date: June 1996 Country of Publication: USA
CODEN: SRECD8 ISSN: 0163-5808
SICI: 0163-5808(199606)25:2L.115:LRSP;1-6
Material Identity Number: A660-96002
U.S. Copyright Clearance Center Code: 0 89791 794 4/96/0006.$3.50
Conference Title: 1996 ACM SIGMOD International Conference on Management of Data
Conference Sponsor: ACM
Conference Date: 4-6 June 1996 Conference Location: Montreal, Que., Canada
Language: English Document Type: Conference Paper (PA); Journal Paper (JP)
Treatment: Practical (P)
Abstract: We present an efficient method to do online reorganization of sparsely populated B/sup +/ trees. It reorganizes the **leaves** first, compacting in short operations groups of **leaves** with the same parent. After compacting, optionally, the new **leaves** may **swap** locations or be moved into **empty** pages so that they are in key **order** on the disk. After the **leaves** are reorganized, the method shrinks the **tree** by making a copy of the upper part of the **tree** while leaving the **leaves** in place. A new concurrency method is introduced so that only a minimum number of pages are locked during reorganization. During **leaf** reorganization, forward recovery is used to save all work already done while maintaining consistency after system crashes. A heuristic algorithm is developed to reduce the number of **swaps** needed during **leaf** reorganization, so that better concurrency and easier recovery can be achieved. A detailed description of switching from the old B/sup +/ **tree** to the new B/sup +/ **tree** is described for the first time. (15 Refs)
Subfile: C
Descriptors: concurrency control; **distributed** databases; **tree data structures** ; trees (mathematics
Identifiers: online reorganization; sparsely populated B/sup +/ trees; **empty** pages; key **order** ; concurrency method; **leaf** reorganization; forward recovery; consistency; heuristic algorithm
Class Codes: C6120 (File organisation); C1160 (Combinatorial mathematics); C6160B (Distributed databases); C6150J (Operating systems)

15/5/43 (Item 11 from file: 34)
DIALOG(R)File 34:SciSearch(R) Cited Ref Sci

04141340 Genuine Article#: RH137 Number of References: 6
Title: THE SIZE OF K-PSEUDOTREES
Author(s): KNILL E; EHRENFEUCHT A; HAUSSLER D
Corporate Source: LOS ALAMOS NATL LAB/LOS ALAMOS//NM/87545; UNIV COLORADO/BOULDER//CO/80309; UNIV CALIF SANTA CRUZ/SANTA CRUZ//CA/95064
Journal: DISCRETE MATHEMATICS, 1995, V141, N1-3 (JUN 28), P185-194
ISSN: 0012-365X
Language: ENGLISH Document Type: ARTICLE
Geographic Location: USA
Subfile: SciSearch
Journal Subject Category: MATHEMATICS
Abstract: Let X be a finite set. A k-pseudotree on X is a family F of subsets of X such that: (i) X is an element of F and for every x is an element of X, {x} is an element of F;(ii) for every U is an element of F there exists an x is an element of U such that if V is an element of F and X is an element of V, then V is comparable to U; (iii) the **intersection** of k + 1 pairwise incomparable members of F is **empty** . The covering graphs of the 1-pseudotrees on an n-set (considered as posets under inclusion) are the directed rooted trees with n **leaves** and no vertex of outdegree one. It is shown that if k < n, then the maximum cardinality of a k-pseudotree on an n-element set is (k + 1)n -((k + 1)k)/2.
Research Fronts: 93-1159 001 (SIMPLE LINEAR TIME ALGORITHM FOR TRIANGULATING 3-COLORED GRAPHS; HYPEREDGE REPLACEMENT; MONADIC 2ND-**ORDER** LOGIC; **TREE** AUTOMATA)
Cited References:
DEWDNEY AK, 1974, V17, P160, J COMBIN THEORY SE B
GRIMALDI RP, 1989, DISCRETE COMBINATORI
JOHNSON DS, 1985, V6, P434, J ALGORITHM
KNILL E, 1991, THESIS U COLORADO BO
SIMONOVITS M, 1980, V6, P301, ANN DISCRETE MATH
WHITEHEAD EG, 1988, V72, P391, DISCRETE MATH

```
Set     Items   Description
S1      66640   TREE OR TREES OR BTREE OR DIRECTORY OR DIRECTORIES OR TRIE
             OR TRIES
S2     656137   NODE? OR BRANCH? OR LEAF? OR JUNCTION? OR JUNCTURE? OR INT-
             ERSECT?
S3      17727   S2(2N)(ADJOIN? OR NEXT OR PRIOR OR PREVIOUS OR FOLLOWING OR
              PARALLEL OR CONTIGUOUS? OR CONNECTING? OR PARENT()CHILD? OR -
             ORDINATE(N)SUBORDINAT? OR LINKED OR SEQUENTIAL?)
S4     115664   EMPTY? OR UNUSED? OR UNFILLED OR "NOT"(N)(FULL OR USED OR -
             USE OR FILLED)
S5    3924933   REARRANG? OR REORDER? OR RESORT? OR REDISTRIBUT? OR DISTRI-
             BUT? OR INSERT OR INSERTING OR INSERTS OR ORDER? OR ARRANG? OR
              SWAP? OR REVERS?
S6     682829   DATASTRUCTUR? OR DATA()(ELEMENTS OR OBJECT OR OBJECTS OR S-
             TRUCTUR? OR ITEMS) OR STACK? OR ARRAY? OR TREE OR BTREE OR MA-
             TRIX?
S7    3024338   END OR ENDPOINT OR ENDS OR TERNIMAL OR TERMINUS OR LEAF? OR
              LEAVES OR TAIL OR TAILS
S8          5   S3 AND S4 AND S5 AND S6
S9        485   S3 AND S5 AND S6
S10       155   S9 AND IC=G06F
S11       203   S2(2N)S4
S12     90123   (S1 OR S2)(2N)S7
S13        21   S10 AND (S11 OR S12)
S14        24   S8 OR S13
S15        19   S14 NOT AD=20010801:20040801
S16        19   S15 NOT AD=20040801:20050901
S17         3   S1 AND S3 AND S5(2N)S6 AND S7
S18        18   S3 AND S5(2N)S6 AND S7
S19        17   (S17 OR S18) NOT S14
S20         2   S19 AND IC=G06F
S21        16   S2(2N)S4(4N)S5
S22         3   S21 AND IC=G06F
S23       796   S1 AND S2 AND S5 AND S6 AND S7
S24       114   S23 AND IC=(G06F-017 OR G06F-007)
S25         4   S4 AND S24
S26     73415   S2(2N)(PAIR? OR TWIN? OR BOTH OR TWO OR SECOND OR 2ND OR A-
             NOTHER OR OTHER OR DUAL OR DUO OR PARALLEL)
S27         4   S25 AND (S4 OR S7) AND S5
S28         0   S27 NOT S25
File 347:JAPIO Nov 1976-2005/Apr(Updated 050801)
         (c) 2005 JPO & JAPIO
File 350:Derwent WPIX 1963-2005/UD,UM &UP=200555
         (c) 2005  Thomson Derwent
```

16/5/2 (Item 2 from file: 347)
DIALOG(R)File 347:JAPIO

05878890 **Image available**
DATA INTEGRATING PROCEDURE DETERMINING METHOD, AND MANUFACTURE PROCEDURE DETERMINING MEANS FOR ARTICLES APPLYING THE SAME

PUB. NO.: 10-161990 [JP 10161990 A]
PUBLISHED: June 19, 1998 (19980619)
INVENTOR(s): MUNAKATA KOICHI
APPLICANT(s): MITSUBISHI ELECTRIC CORP [000601] (A Japanese Company or Corporation), JP (Japan)
APPL. NO.: 08-324034 [JP 96324034]
FILED: December 04, 1996 (19961204)
INTL CLASS: [6] **G06F-017/00**
JAPIO CLASS: 45.4 (INFORMATION PROCESSING -- Computer Applications)

ABSTRACT

PROBLEM TO BE SOLVED: To generate a process representation **tree** in a short time by generating the process representation **tree** representing an executable manufacture procedure by modifying a dependency graph obtained from an initial graph.

SOLUTION: An initial essential node selecting means of step ST11 in step ST1 selects a node, which is a final node among supply nodes or process nodes and outputs a necessary component without fail, the selected node is regarded as an initial essential node, and a **node connecting** means of step ST12 generates a variable table. Nodes are generated corresponding to respective element processes except an **end node** . A directional branch is generated from a node corresponding to a process which selects variables in **order** and uses the variables as output variables to a node corresponding to a process which uses the variables as input variables. A dependence graph generating means of a step ST2 modifies the initial graph into the dependency graph and a process representation **tree** generating procedure of step ST3 modifies the dependency graph into the process representation **tree** representing the executable manufacture procedure.

16/5/5 (Item 1 from file: 350)
DIALOG(R)File 350:Derwent WPIX

015020218 **Image available**
WPI Acc No: 2003-080735/200308
XRPX Acc No: N03-063096

Network address search method involves reading table flag which shows following node **or** leaf **information on** node **of search** tree **, using which** node **or** leaf **table which pointer points is chosen and address search is completed**

Patent Assignee: MITSUBISHI ELECTRIC CORP (MITQ)
Number of Countries: 001 Number of Patents: 001
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
JP 2002290447	A	20021004	JP 200190170	A	20010327	200308	B

Priority Applications (No Type Date): JP 200190170 A 20010327
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
JP 2002290447	A		20	H04L-012/56	

Abstract (Basic): JP 2002290447 A

NOVELTY - An entry is read in **order** from head bit of the network address included in receiving packet. A node table has table pointer for reading the table flag which shows **following node** or **following leaf** information on the node of the search **tree** . The **node** table or **leaf** table to which the pointer points is chosen and address search is completed, based on the following table flag.

DETAILED DESCRIPTION - INDEPENDENT CLAIMS are included for the following:

(1) Address search circuit; and

(2) Address search program.

USE - For searching network address.

ADVANTAGE - The formation of search table reduces the memory consumption, and the address search speed is improved.

DESCRIPTION OF DRAWING(S) - The figure shows a structural diagram of the relay device realizing address search processing. (Drawing includes non-English language text).

pp; 20 DwgNo 1/15

Title Terms: NETWORK; ADDRESS; SEARCH; METHOD; READ; TABLE; FLAG; SHOW; FOLLOW; NODE; **LEAF** ; INFORMATION; NODE; SEARCH; **TREE** ; NODE; **LEAF** ; TABLE; POINT; POINT; CHOICE; ADDRESS; SEARCH; COMPLETE
Derwent Class: T01; W01
International Patent Class (Main): H04L-012/56
International Patent Class (Additional): **G06F-017/30**
File Segment: EPI

16/5/7 (Item 3 from file: 350)
DIALOG(R)File 350:Derwent WPIX

013330693 **Image available**
WPI Acc No: 2000-502632/200045
XRPX Acc No: N00-372889

Multidimensional space data structure **has node pointer which shows storing position of multidimensional partial space symbol showing starting point and terminus of number of entries**

Patent Assignee: NIPPON TELEGRAPH & TELEPHONE CORP (NITE)
Number of Countries: 001 Number of Patents: 002
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
JP 2000200342	A	20000718	JP 991345	A	19990106	200045	B
JP 3542732	B2	20040714	JP 991345	A	19990106	200446	

Priority Applications (No Type Date): JP 991345 A 19990106
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
JP 2000200342	A		15	G06T-001/00	
JP 3542732	B2		20	G06T-001/00	Previous Publ. patent JP 2000200342

Abstract (Basic): JP 2000200342 A

NOVELTY - A node pointer shows the storing position of a multidimensional partial space symbol showing the starting point and the terminus of the number of entries. The number of entries pertains to the nodes joined by a virtual portion. A virtual range rectangle is relatively expressed with the partial space symbol based on a position to the minimum range rectangle in a non- **leaf node** .

DETAILED DESCRIPTION - A geometric object in multidimensional space is packed to minimum range rectangle. A **tree** structure which joined the **leaf node sequentially** to the lower **order** of the non- **leaf node** is provided. An absolute position is expressed as the minimum range rectangle and is considered as an actual unit. INDEPENDENT CLAIMS are also included for the following:

(a) a multidimensional space **data structure** updating method;
(b) a multidimensional space **data structure** search procedure;
(c) and a recording medium.

USE - None given.

ADVANTAGE - Enables detection of an object using less disc access in search process, thus improving search capability.

DESCRIPTION OF DRAWING(S) - The figure shows the diagram of the multidimensional space **data structure** .

pp; 15 DwgNo 4/12

Title Terms: MULTIDIMENSIONAL; SPACE; DATA; STRUCTURE; NODE; POINT; SHOW; STORAGE; POSITION; MULTIDIMENSIONAL; SPACE; SYMBOL; START; POINT; TERMINAL; NUMBER; ENTER
Derwent Class: T01
International Patent Class (Main): G06T-001/00
International Patent Class (Additional): **G06F-017/30**
File Segment: EPI

16/5/8 (Item 4 from file: 350)
DIALOG(R)File 350:Derwent WPIX

012990924 **Image available**
WPI Acc No: 2000-162776/200015
XRPX Acc No: N00-121550

Storage and organization method for multimedia objects by decomposing multimedia tree **structure having multiple nodes into graph map using single-link graph nodes**

Patent Assignee: MATSUSHITA ELECTRIC IND CO LTD (MATU)
Inventor: CHEN J; NG K L; TAN P Y
Number of Countries: 025 Number of Patents: 001
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
EP 977128	A1	20000202	EP 98114113	A	19980728	200015	B

Priority Applications (No Type Date): EP 98114113 A 19980728
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
EP 977128	A1	E	18	G06F-017/30	

Designated States (Regional): AL AT BE CH CY DE DK ES FI FR GB GR IE IT LI LT LU LV MC MK NL PT RO SE SI

Abstract (Basic): EP 977128 A1

NOVELTY - The method for storing and decomposing multimedia objects involves decomposing a multilevel **tree** structure into a graph map using single-link graph nodes. A name or address is assigned to each node in the **tree** structure in ascending **order** from a root node to the last **leaf node**, and each node in the single-link graph node is mapped to a single-link graph node having a pointer memory. The depth or breadth coupled graph nodes are stored in page memory.

DETAILED DESCRIPTION - INDEPENDENT CLAIMS are included for; a method of packing **sequentially ordered** graph **nodes** into a page memory; a method that enables organizing and administering of groups of multimedia objects organized in a **tree** structure; a method for performing breadth-first multimedia objects search and retrieval.

USE - Storage, retrieval and organization of multimedia objects eg. Synchronized audio and video information, audio and video in synchronised form and general data and executable codes in a **distributed** and federate manner.

ADVANTAGE - Provides efficient data storage and retrieval while retaining hierarchy of a **tree** . Facilitates searching and improves multimedia object retrieval.

DESCRIPTION OF DRAWING(S) - The drawing shows a flow chart of **tree** nodes to graph nodes mapping.

pp; 18 DwgNo 2/7

Title Terms: STORAGE; METHOD; OBJECT; DECOMPOSE; **TREE** ; STRUCTURE; MULTIPLE; NODE; GRAPH; MAP; SINGLE; LINK; GRAPH; NODE
Derwent Class: T01
International Patent Class (Main): **G06F-017/30**
File Segment: EPI

16/5/9 (Item 5 from file: 350)
DIALOG(R)File 350:Derwent WPIX

012471024 **Image available**
WPI Acc No: 1999-277132/199923
XRPX Acc No: N99-207761

Method of organizing multilevel memory structure with upper root end **and lower** nodes **and** branches **representing** parallel **data fields to be compressed eliminates fields at each lower branch or node**

Patent Assignee: TRIADA LTD (TRIA-N)
Inventor: BUGAJSKI J M; RAGHAVAN K R; ZHANG T
Number of Countries: 082 Number of Patents: 005
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
WO 9917231	A1	19990408	WO 98US18970	A	19980910	199923	B
AU 9894789	A	19990423	AU 9894789	A	19980910	199935	
US 5966709	A	19991012	US 97939023	A	19970926	199949	
EP 1016008	A1	20000705	EP 98948158	A	19980910	200035	
			WO 98US18970	A	19980910		
JP 2001518726	W	20011016	WO 98US18970	A	19980910	200176	
			JP 2000514226	A	19980910		

Priority Applications (No Type Date): US 97939023 A 19970926
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
WO 9917231	A1	E	20	G06F-017/30	

Designated States (National): AL AM AT AU AZ BA BB BG BR BY CA CH CN CU CZ DE DK EE ES FI GB GE GH GM HU ID IL IS JP KE KG KP KR KZ LC LK LR LS LT LU LV MD MG MK MN MW MX NO NZ PL PT RO RU SD SE SG SI SK SL TJ TM TR TT UA UG UZ VN YU ZW
Designated States (Regional): AT BE CH CY DE DK EA ES FI FR GB GH GM GR IE IT KE LS LU MC MW NL OA PT SD SE SZ UG ZW

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
AU 9894789	A				Based on patent WO 9917231
US 5966709	A			G06F-017/30	
EP 1016008	A1	E		G06F-017/30	Based on patent WO 9917231

Designated States (Regional): AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
JP 2001518726	W		20	H03M-007/30	Based on patent WO 9917231

Abstract (Basic): WO 9917231 A1

NOVELTY - The method after making parallel streams' initial list **orders** them based on increasing cardinality. Adjacent nodes are paired (106) and children of resulting node eliminated from list, and new parent node is added to list. The new list is **rearranged** from right to left as a function of increasing cardinality. The pairing steps are repeated until a single root node remains for the final memory.

USE - For analyzing the cardinality of input data for the purpose of optimizing the **ordered data structure** .

ADVANTAGE - Automatically determines the optimum multilevel memory structure to maximize compression within a NGRAM environment.

DESCRIPTION OF DRAWING(S) - The drawing shows a flowchart of the overall method.

pairing of the nodes (106)

pp; 20 DwgNo 1/4

Title Terms: METHOD; MULTILEVEL; MEMORY; STRUCTURE; UPPER; ROOT; END; LOWER ; NODE; BRANCH; REPRESENT; PARALLEL; DATA; FIELD; COMPRESS; ELIMINATE; FIELD; LOWER; BRANCH; NODE
Derwent Class: T01
International Patent Class (Main): **G06F-017/30** ; H03M-007/30
International Patent Class (Additional): **G06F-007/00**
File Segment: EPI

16/5/10 (Item 6 from file: 350)
DIALOG(R)File 350:Derwent WPIX

010620330 **Image available**
WPI Acc No: 1996-117283/199612
XRPX Acc No: N96-098021

Efficient string searching method for data compression system - involves encoding and decoding user data using linked data structure **and maintaining linked list** data structure **for input characters**

Patent Assignee: MOTOROLA INC (MOTI)
Inventor: DEMELLO W M; FULLING F
Number of Countries: 020 Number of Patents: 009
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
WO 9603809	A1	19960208	WO 95US6349	A	19950522	199612	B
EP 721699	A1	19960717	EP 95920557	A	19950522	199633	
			WO 95US6349	A	19950522		
FI 9601369	A	19960325	WO 95US6349	A	19950522	199635	
			FI 961369	A	19960325		
US 5564045	A	19961008	US 94281946	A	19940728	199646	
CN 1131480	A	19960918	CN 95190683	A	19950522	199801	
EP 721699	B1	20030423	EP 95920557	A	19950522	200329	
			WO 95US6349	A	19950522		
DE 69530470	E	20030528	DE 95630470	A	19950522	200343	
			EP 95920557	A	19950522		
			WO 95US6349	A	19950522		
FI 115350	B1	20050415	WO 95US6349	A	19950522	200526	
			FI 961369	A	19960325		
CN 1097881	C	20030101	CN 95190683	A	19950522	200532	

Priority Applications (No Type Date): US 94281946 A 19940728
Cited Patents: US 5058144; US 5151697; US 5239298
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
WO 9603809	A1	E	25	H03M-007/30	
Designated States (National): CA CN FI					
Designated States (Regional): AT BE CH DE DK ES FR GB GR IE IT LU MC NL PT SE					
EP 721699	A1	E	29	H03M-007/30	Based on patent WO 9603809
Designated States (Regional): DE FR GB					
FI 9601369	A			H03M-000/00	
US 5564045	A		9	G06F-017/30	
CN 1131480	A			H03M-007/30	
EP 721699	B1	E		H03M-007/30	Based on patent WO 9603809
Designated States (Regional): DE FR GB					
DE 69530470	E			H03M-007/30	Based on patent EP 721699
					Based on patent WO 9603809
FI 115350	B1			H03M-007/30	Previous Publ. patent FI 9601369
CN 1097881	C			H03M-007/30	

Abstract (Basic): WO 9603809 A

The method involves **inserting** a termination node which is initially linked to every root node. The termination node is used for at least one of the following: determining a potential match in a child list, adding a new **node prior** to the termination **node** , determining a **next leaf node** and, where the **next leaf node** is the termination node, recycling to a start of the **data structure** . Finally if the **next leaf node** is a non-termination **node** , the **leaf node** is deleted **prior** to the termination node.

The method further involves string searching for efficiently using and maintaining a linked list **data structure** for input characters.

A termination node is inserted in a memory unit where the termination **node** is initially **linked** to every root node of the

memory unit. A termination node-based novel scheme is used for simplifying a string-searching process in a processor such that processing is minimized and throughput performance is maximized.

ADVANTAGE - Provides efficient data storage and fast data communication. Processor use for computation is minimised, throughput delay is minimised and throughput performance is maximised.

Dwg.1/3

Title Terms: EFFICIENCY; STRING; SEARCH; METHOD; DATA; COMPRESS; SYSTEM; ENCODE; DECODE; USER; DATA; LINK; DATA; STRUCTURE; MAINTAIN; LINK; LIST; DATA; STRUCTURE; INPUT; CHARACTER

Derwent Class: T01; U21; W01

International Patent Class (Main): **G06F-017/30** ; H03M-000/00; H03M-007/30

International Patent Class (Additional): **G06F-013/38** ; H04B-001/38

File Segment: EPI

16/5/11 (Item 7 from file: 350)
DIALOG(R)File 350:Derwent WPIX

010543530 **Image available**
WPI Acc No: 1996-040484/199604
XRPX Acc No: N96-034013

Storing and retrieving data and memory arrangement **esp for use in telephone exchange database - using search key in directory structure having nodes at several different levels and minimising space required for storage of data**

Patent Assignee: NOKIA TELECOM OY (OYNO); NOKIA NETWORKS OY (OYNO)
Inventor: TIKKANEN M
Number of Countries: 064 Number of Patents: 010
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
WO 9534155	A2	19951214	WO 95FI319	A	19950605	199604	B
AU 9526174	A	19960104	AU 9526174	A	19950605	199613	
WO 9534155	A3	19960208	WO 95FI319	A	19950605	199622	
EP 772836	A1	19970514	EP 95920913	A	19950605	199724	
			WO 95FI319	A	19950605		
JP 10504407	W	19980428	WO 95FI319	A	19950605	199827	
			JP 96500403	A	19950605		
AU 690282	B	19980423	AU 9526174	A	19950605	199828	
US 5848416	A	19981208	WO 95FI319	A	19950605	199905	
			US 96750777	A	19961206		
CN 1152365	A	19970618	CN 95194055	A	19950605	200132	
EP 772836	B1	20011212	EP 95920913	A	19950605	200204	
			WO 95FI319	A	19950605		
DE 69524601	E	20020124	DE 624601	A	19950605	200215	
			EP 95920913	A	19950605		
			WO 95FI319	A	19950605		

Priority Applications (No Type Date): FI 942664 A 19940606; FI 942663 A 19940606
Cited Patents: EP 650131; US 5319777; No-SR.Pub
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
WO 9534155	A2	E	35	H04M-000/00	

Designated States (National): AM AT AU BB BG BR BY CA CH CN CZ DE DK EE ES FI GB GE HU IS JP KE KG KP KR KZ LK LR LT LU LV MD MG MN MW MX NO NZ PL PT RO RU SD SE SG SI SK TJ TM TT UA UG US UZ VN
Designated States (Regional): AT BE CH DE DK ES FR GB GR IE IT KE LU MC MW NL OA PT SD SE SZ UG

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
AU 9526174	A			H04M-001/00	Based on patent WO 9534155
WO 9534155	A3			H04M-000/00	
EP 772836	A1	E		G06F-017/30	Based on patent WO 9534155

Designated States (Regional): AT BE DE FR GB IT NL SE

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
JP 10504407	W		30	G06F-017/30	Based on patent WO 9534155
AU 690282	B			H04M-001/00	Previous Publ. patent AU 9526174
					Based on patent WO 9534155
US 5848416	A			G06F-017/30	Based on patent WO 9534155
CN 1152365	A			G06F-017/30	
EP 772836	B1	E		G06F-017/30	Based on patent WO 9534155

Designated States (Regional): AT BE DE FR GB IT NL SE

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
DE 69524601	E			G06F-017/30	Based on patent EP 772836
					Based on patent WO 9534155

Abstract (Basic): WO 9534155 A

The method for storing data identifiable by search key in memory involves selecting from the search key related to each dimension a predetermined dimension specific number of bits and using them to form a search word on the basis of which the **next node** is sought from the internal node at the root level of the **tree** shaped hierarchy. A predetermined dimension specific number of bits are selected from the

unselected bits in the search key related to each dimension. They are used to form a search word with which the address of a further new node at a lower level is sought from the **array** of the node that has been accessed.

The process is repeated until an **empty** element has been encountered or until the address of the new node at a lower level is the address of a **leaf node** . A pointer is stored in the **leaf node** and the data unit at the storage location is indicated by the pointer.

USE/ADVANTAGE - Maintenance of subscriber database in telephone exchange. Requires less hardware. Permits partial key retrieval.

Dwg.4/4

Title Terms: STORAGE; RETRIEVAL; DATA; MEMORY; **ARRANGE** ; TELEPHONE; EXCHANGE; DATABASE; SEARCH; KEY; DIRECTORY; STRUCTURE; NODE; LEVEL; MINIMISE; SPACE; REQUIRE; STORAGE; DATA

Derwent Class: T01; W01

International Patent Class (Main): **G06F-017/30** ; H04M-000/00; H04M-001/00

International Patent Class (Additional): H04Q-003/545; H04Q-003/76

File Segment: EPI

25/5/3 (Item 3 from file: 350)
DIALOG(R)File 350:Derwent WPIX

012813653 **Image available**
WPI Acc No: 1999-619884/199953
XRPX Acc No: N99-457173

Spatially similar high dimensional data object **points associating method for database applications**

Patent Assignee: INT BUSINESS MACHINES CORP (IBMC)
Inventor: AGRAWAL R; SHIM K; SRIKANT R
Number of Countries: 001 Number of Patents: 001
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
US 5978794	A	19991102	US 96629688	A	19960409	199953	B

Priority Applications (No Type Date): US 96629688 A 19960409
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
US 5978794	A		14	G06F-017/00	

Abstract (Basic): US 5978794 A

NOVELTY - The points associated with the pair of **leaf node** selected by scanning of interior **nodes** of **data structure** , are sort-merged, based on the common sort dimension. The points of selected pair of **leaf node** are joined, when distance between any two points is at most epsilon.

DETAILED DESCRIPTION - A multi-dimensional **data structure** having several **leaf nodes** for organizing the points, in created. Each **leaf node** is split into (1/epsilon) child **nodes** , where epsilon is similar distance, based on the depth of the **leaf node** . When the number of points associated with the **leaf node** exceeds a predetermined value, the dimensions used for splitting the **nodes** in an **order** of correlation among the dimensions, such that the dimension next to the dimension used for splitting has the least correlation with previously used dimensions. The points in each **leaf node** is sorted using one of the dimensions **not used** for splitting the **leaf nodes** , as common sort dimension. INDEPENDENT CLAIMS are also included for the following:

(a) high dimensional **data object** points associating system;

(b) a program product for associating high dimensional **data object** points

USE - For coupling spatially similar dimensional **data objects** in multi-media database, scientific database, medical database, time series database.

ADVANTAGE - Since the **order** of dimensions to be split is determined based on correlations between the dimensions, the system storage requirements during coupling operator is minimized greatly. The use of the common sort dimension eliminates the need for repeatedly sorting the points during coupling operation. Since the global **ordering** is used for selecting the split dimensions, the number of neighbor **nodes** to be examined are minimized. Since algorithms are offered for generating the E-K-D- B **tree** using biased splitting, the number of **nodes** to be examined during coupling operation are reduced.

DESCRIPTION OF DRAWING(S) - The figure shows flowchart illustrating the overall operations involved in spatially similar high dimensional **data objects** coupling method.

pp; 14 DwgNo 1/8

Title Terms: SPACE; SIMILAR; HIGH; DIMENSION; DATA; OBJECT; POINT; ASSOCIATE; METHOD; DATABASE; APPLY
Derwent Class: T01
International Patent Class (Main): **G06F-017/00**
File Segment: EPI

```
Set      Items   Description
S1       66640   TREE OR TREES OR BTREE OR DIRECTORY OR DIRECTORIES OR TRIE
              OR TRIES
S2      656137   NODE? OR BRANCH? OR LEAF? OR JUNCTION? OR JUNCTURE? OR INT-
              ERSECT?
S3         349   S1(2N)(ADJOIN? OR CONTIGUOUS? OR CONNECTING? OR PARENT()CH-
              ILD? OR ORDINATE(N)SUBORDINAT? OR LINKED OR SEQUENTIAL?)
S4      115664   EMPTY? OR UNUSED? OR UNFILLED OR "NOT"(N)(FULL OR USED OR -
              USE OR FILLED)
S5     3924933   REARRANG? OR REORDER? OR RESORT? OR REDISTRIBUT? OR DISTRI-
              BUT? OR INSERT OR INSERTING OR INSERTS OR ORDER? OR ARRANG? OR
               SWAP? OR REVERS?
S6      682829   DATASTRUCTUR? OR DATA()(ELEMENTS OR OBJECT OR OBJECTS OR S-
              TRUCTUR? OR ITEMS) OR STACK? OR ARRAY? OR TREE OR BTREE OR MA-
              TRIX?
S7     3024338   END OR ENDPOINT OR ENDS OR TERNIMAL OR TERMINUS OR LEAF? OR
               LEAVES OR TAIL OR TAILS
S8          86   S3 AND S5
S9          72   S8 AND (S4 OR S6 OR S7)
S10          0   S4 AND S8
S11        117   S2 AND S4 AND S5 AND S6 AND (S7 OR S4)
S12          1   S3 AND S4 AND (S5 OR S6)
S13         17   S8 AND S6 AND S7
S14         18   S2(N)S4 AND S5
S15          3   S1 AND S14
S16         20   S15 OR S13
S17         20   IDPAT (sorted in duplicate/non-duplicate order)
S18         20   IDPAT (primary/non-duplicate records only)
File 347:JAPIO Nov 1976-2005/Apr(Updated 050801)
         (c) 2005 JPO & JAPIO
File 350:Derwent WPIX 1963-2005/UD,UM &UP=200555
         (c) 2005  Thomson Derwent
```

12/5/1 (Item 1 from file: 350)
DIALOG(R)File 350:Derwent WPIX

009491198 **Image available**
WPI Acc No: 1993-184733/199323
Related WPI Acc No: 1990-356929; 1993-184732; 1993-184734
XRPX Acc No: N93-141951

Re-configurable signal processor - realises generic capability for fault-tolerant and re-configurable multiprocessor computer scalable to thousands of processor elements

Patent Assignee: AMERICAN TELEPHONE & TELEGRAPH CO (AMTT)
Inventor: GORIN A L; MAKOFSKY P A; MORTON N; OLIVER N C; SHIVELY R R; STANSIOLA C A
Number of Countries: 001 Number of Patents: 001
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
GB 2262174	A	19930609	GB 906712	A	19900326	199323	B
			GB 931713	A	19930128		

Priority Applications (No Type Date): US 89331411 A 19890331
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
GB 2262174	A		29	G06F-011/20	Derived from application GB 906712

Abstract (Basic): GB 2262174 A

The apparatus expands a **tree** multiprocessor topology while maintaining a constant number of root connection paths to the topology, and a constant number of expansion nodes, comprising two **arrays** of substantially identical processor elements. Each element having four ports, and selectively connects ports of adjacent ones of all but two of the elements in each **array** , to form in each **array** a two-root sub- **tree** of processor elements.

The two elements **not used** in the sub-trees each have three-port expansion nodes. The two roots and the three-port expansion nodes thereby furnishing eight connection paths to each **array** . A way for **connecting** the sub- **tree** and a first expansion node in the first **array** to the corresponding parts of the second **array** . This forms a further two-root sub- **tree** , the second expansion node of each **array** being available to replace elements in its respective **array** , and two roots of the further sub- **tree** and the last-named nodes thereby having a total of eight connection paths to the combined assemblages of the two processor element **arrays** .

USE/ADVANTAGE - Enables or disables nodes by revising communication path. Adds steps to application program to convey idealised or nominal system configuration.

Dwg.3/16

Title Terms: CONFIGURATION; SIGNAL; PROCESSOR; REALISE; CAPABLE; FAULT; TOLERATE; CONFIGURATION; MULTIPROCESSOR; COMPUTER; THOUSAND; PROCESSOR; ELEMENT
Derwent Class: T01
International Patent Class (Main): G06F-011/20
File Segment: EPI

18/5/5 (Item 5 from file: 350)
DIALOG(R)File 350:Derwent WPIX

013930261 **Image available**
WPI Acc No: 2001-414475/200144
XRPX Acc No: N01-306917

Similar featured-variable search for internet, involves assigning link which is followed so that lower order **node approached from main** directory **, with minimum distance is referred, on** tree **structure index searching**

Patent Assignee: NIPPON TELEGRAPH & TELEPHONE CORP (NITE)
Number of Countries: 001 Number of Patents: 002
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
JP 2001134594	A	20010518	JP 99316327	A	19991108	200144	B
JP 3615439	B2	20050202	JP 99316327	A	19991108	200511	

Priority Applications (No Type Date): JP 99316327 A 19991108
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
JP 2001134594	A		14	G06F-017/30	
JP 3615439	B2		19	G06F-017/30	Previous Publ. patent JP 2001134594

Abstract (Basic): JP 2001134594 A

NOVELTY - The lower **order empty nodes** of an hierarchy, are detected at time of construction of **tree** structure index. A link is assigned and is followed so that the lower **order** node which can be approached from main **directory** with minimum distance is referred, during searching **tree** structure index. Within the leaf node, the nearest neighbor point is searched based on near featured-variable vector.

DETAILED DESCRIPTION - INDEPENDENT CLAIMS are also included for the following:

(a) Similar featured-variable search apparatus;

(b) Recording medium

USE - For searching multimedia data on internet.

ADVANTAGE - Even the intermediate nodes on **tree** structure can be determined easily by this method and similar featured-variable search efficiency is improved.

DESCRIPTION OF DRAWING(S) - The figure shows the components of similar featured-variable search apparatus. (Drawing includes non-English language text).

pp; 14 DwgNo 1/13

Title Terms: SIMILAR; VARIABLE; SEARCH; ASSIGN; LINK; FOLLOW; SO; LOWER; **ORDER** ; NODE; APPROACH; MAIN; **DIRECTORY** ; MINIMUM; DISTANCE; REFER; **TREE** ; STRUCTURE; INDEX; SEARCH
Derwent Class: T01
International Patent Class (Main): G06F-017/30
International Patent Class (Additional): G06T-007/00
File Segment: EPI

18/5/12 (Item 12 from file: 350)
DIALOG(R)File 350:Derwent WPIX

003562997
WPI Acc No: 1983-B1188K/198304
XRPX Acc No: N83-014844

Reduction processor for executing programs stored as tree **-like graphs - has storage device to retrieve two-cell nodes for reduction to produce result through steps of substitutions**

Patent Assignee: BURROUGHS CORP (BURS)
Inventor: BOLTON B C; HAGENMAIER C F; LOGSDON G L; MINER R L
Number of Countries: 008 Number of Patents: 005
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
EP 69313	A	19830112	EP 82105702	A	19820628	198304	B
US 4447875	A	19840508	US 81281064	A	19810707	198421	
CA 1211221	A	19860109				198641	
EP 69313	B	19870812				198732	
DE 3276970	G	19870917				198738	

Priority Applications (No Type Date): US 81281064 A 19810707
Cited Patents: 4.Jnl.Ref; No-SR.Pub
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
EP 69313	A	E	29		

Designated States (Regional): BE DE FR GB IT NL

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
EP 69313	B	E			

Designated States (Regional): BE DE FR GB IT NL

Abstract (Basic): EP 69313 A

The system is for executing programs stored as treelike graphs, employing variable-free applicative language codes. The system comprises a storage device for receiving two-cell nodes representing different graphs. The processor is coupled to the storage device to retrieve the two-cell nodes for reduction to produce a result through one or more steps of a series of substitutions.

A control section is coupled to the data section to provide signals so as to enable the substitution to be performed. The control section includes microcode memory and a control register.

Title Terms: REDUCE; PROCESSOR; EXECUTE; PROGRAM; STORAGE; **TREE** ; GRAPH; STORAGE; DEVICE; RETRIEVAL; TWO; CELL; NODE; REDUCE; PRODUCE; RESULT; THROUGH; STEP; SUBSTITUTION
Derwent Class: T01
International Patent Class (Additional): G06F-007/00; G06F-009/44; G06F-013/16
File Segment: EPI